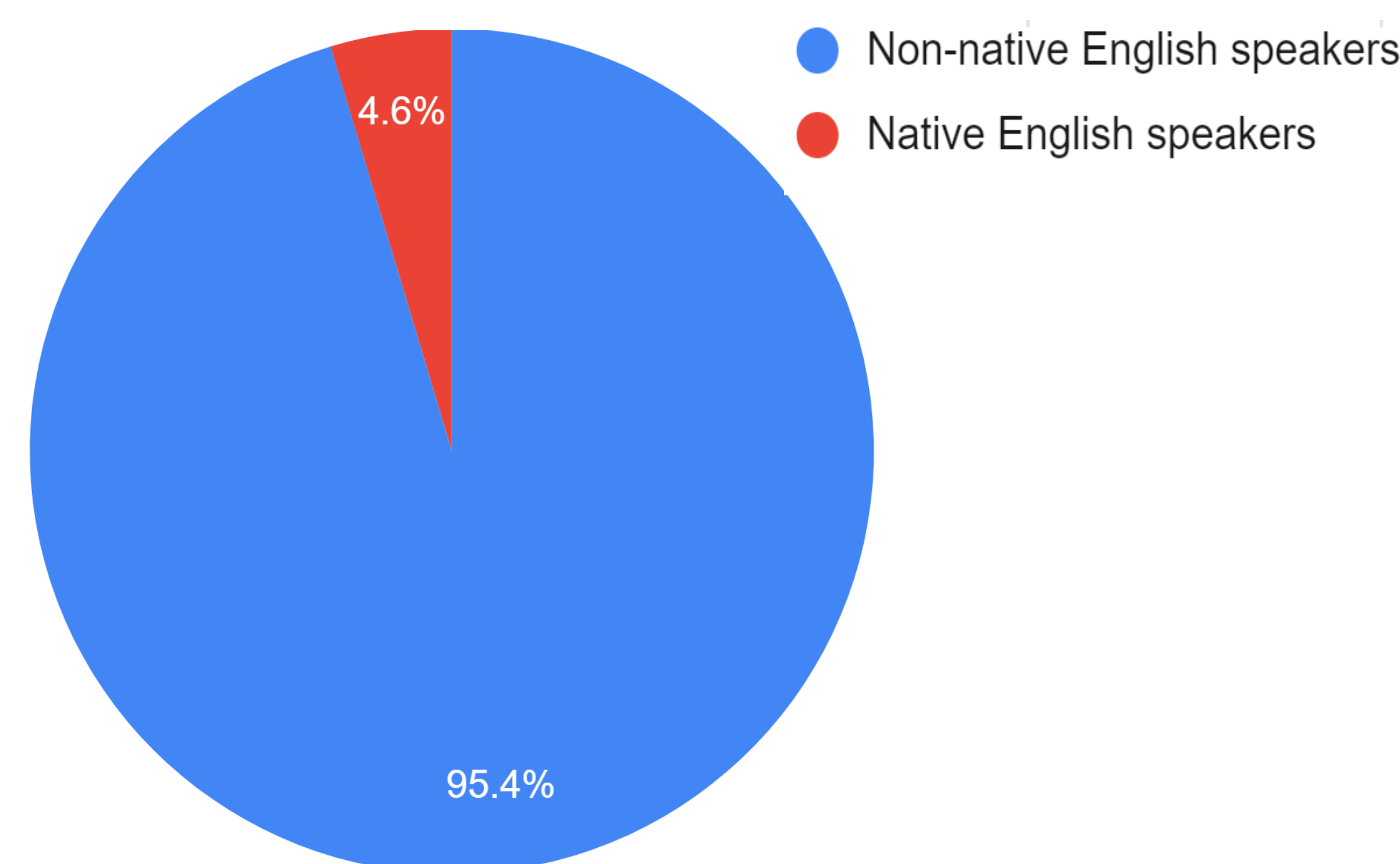


## BACKGROUND

Computer Science is an English-based subject, yet only 5% of the world's population (360 million people out of 7.8 billion people) speak English as their native language. (Source: Ethnologue, 21st edition)



The most popular programming languages are all designed in English.

| English Programming Language                                                                                             | Non - English Programming Language                                                                                                                                            |
|--------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>Java</li> <li>Python</li> <li>C/ C++</li> <li>HTML</li> <li>JavaScript</li> </ul> | <ul style="list-style-type: none"> <li>Wenyan (Chinese)</li> <li>GOTO++ (French)</li> <li>Kalamm.io (Hindi)</li> <li>Produire (Japanese)</li> <li>GarGar (Spanish)</li> </ul> |

## MOTIVATION

Our motivation to do research with non-native English-speaking students (NNES) in Computer Science comes from the need to see if they struggle in a field that is strongly rooted in the English language.

## RESEARCH QUESTIONS

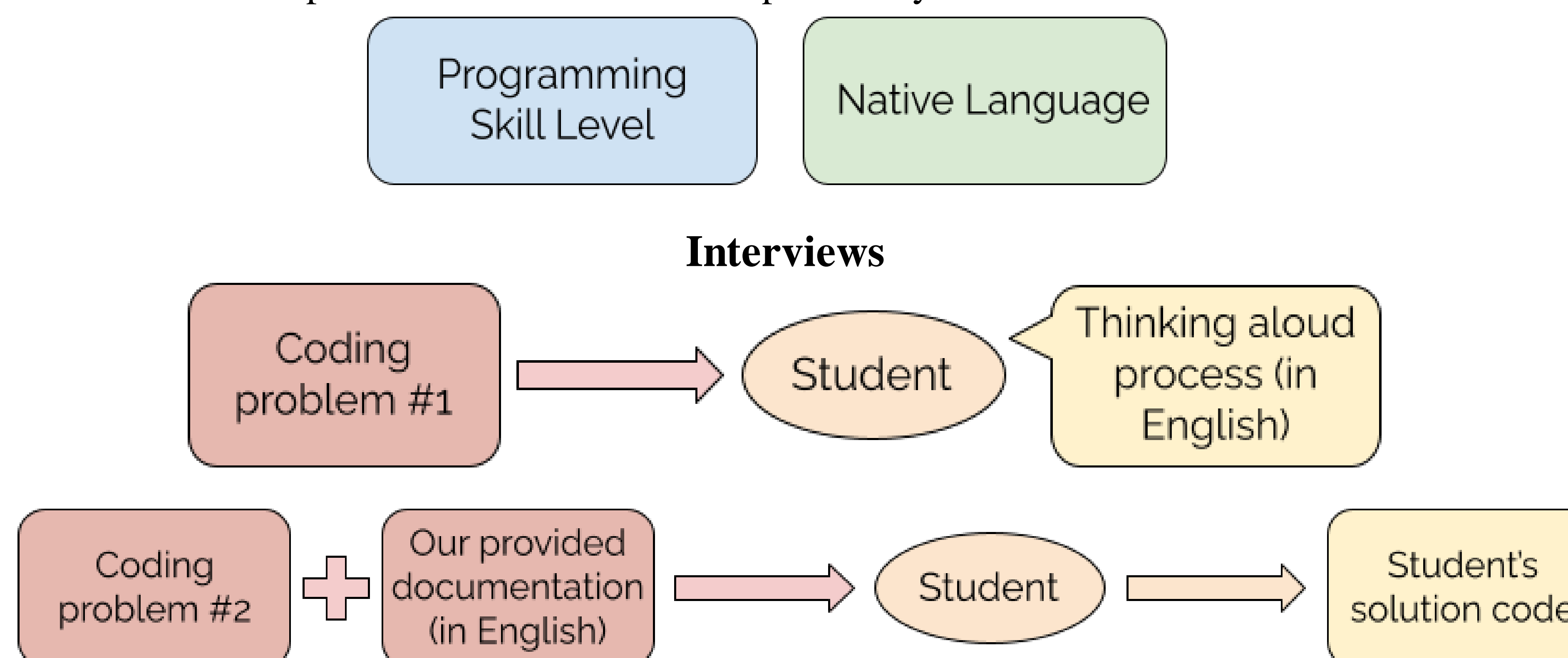
- Are there any challenges that non-native English speakers (NNES) face as they talk through their steps while coding?
- Is there a difference in how NNES read and understand English programming documentation and whether this affects their coding performance compared to native English-speaking students (NES)?

## METHOD

We conducted a study with 4 NNES and 3 NES. The different components of the study provide different data.

### Pre-survey

We collected two pieces of information in our pre-survey:

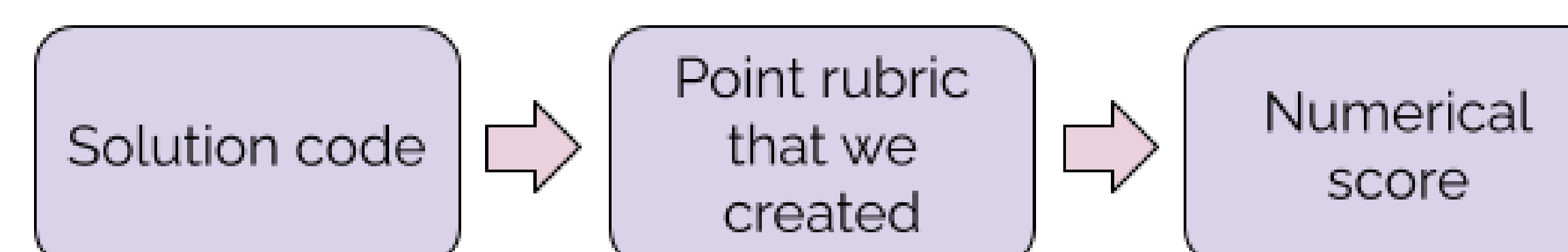


### Post Interview Questions

The Post-Interview allowed the students to reflect on the interviews that they participated in. We asked questions to understand whether the students felt like they struggled and if they did, what factors played into those struggles.

## ANALYSIS

Quantitative: (process was done twice separately, one for each interview)

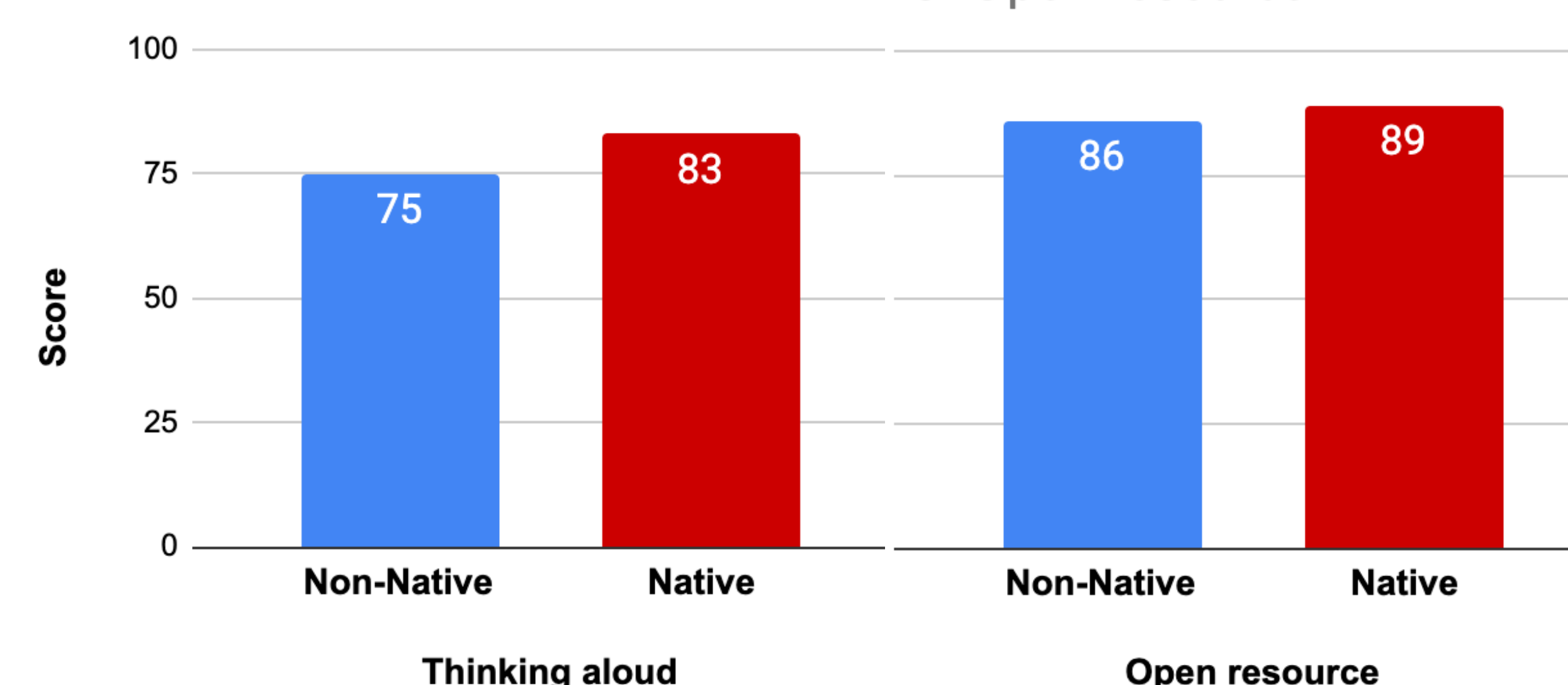


Qualitative: (Grounded Theory analysis)



## QUANTITATIVE RESULTS

Total score percentage for native vs non-native students on 2 coding problems



## QUALITATIVE RESULTS

### LOGIC

- Thinking-aloud interview:** Most of the participants had a hard time with the logic of prime numbers.
- Open-resource interview:** All students understood the logic and utilized the step-by-step instructions.

### IMPLEMENTATION

- Thinking-aloud interview:** Both NES and NNES struggled to translate prime numbers logic into code.
- Open-resource interview:** Both NES and NNES had difficulties debugging their solution code, specifically errors regarding Python file handling.

### THOUGHTS

- Thinking-aloud interview:** NES and NNES prefer either thinking-aloud or other forms of thinking (i.e., thinking in their head, brainstorming on paper, etc.)
- Open-resource interview:** NES and NNES found the Unicode documentation helpful but not the official Python documentation. The students either preferred class materials or online resources.

### FEELINGS

- Thinking-aloud interview:** NES and NNES felt lack of confidence about the thinking-aloud problem.
- Open-resource interview:** NES and NNES felt lack of confidence about their Python programming skills.

## CONCLUSION

- Some students did face challenges while thinking aloud, but they were not limited to NNES specifically.
- NES and NNES expressed similar thoughts about the documentation; there is no difference in how they read and understand English programming documentation.

Based on our research, we recommend that:

- Tech companies may need to rethink how they conduct their think-aloud interviews.
- We need more beginner-friendly documentation since the current official documentation target professional users.

## ACKNOWLEDGEMENTS

We would like to thank our advisor Adalbert Gerald Soosai Raj, Professor Christine Alvarado and Vignesh Gokul for their guidance. We would also like to thank the ERSP program for providing us with this opportunity.